

הנחיות כלליות

- המתרגל האחראי על תרגיל זה הוא: **אייל**.
 - יש לשלוח את הקבצים באמצעות **מערכת ההגשה** לפני חלוף התאריך **29/11/21**.
 - ניתן להגיש את התרגיל באיחור עם קנס אוטומטי על פי הפירוט הבא:
 - 👁️ יום איחור (**30/11/21**) - קנס של **10 נקודות** (ציון מקסימלי – 90).
 - 👁️ יומיים איחור (**1/12/21**) - קנס של **20 נקודות** (ציון מקסימלי – 80).
 - 💣 לאחר מכן לא יהיה ניתן להגיש את התרגיל (ציון 0).
 - שאלות בנוגע לתרגיל יש לפרסם **באופן ציבורי בפורום הקורס** בלבד!
 - בקשות להארכה (מסיבות מוצדקות בלבד, כגון: מילואים, לידה וכו') יש לפרסם **באופן פרטי בפורום הקורס** בלבד (יש למען את הפוסט ל-instructors).
 - בכל בקשה יש לציין:
 - 👤 שם מלא.
 - 💻 שם משתמש במערכת ההגשה.
 - 💾 תעודת זהות.
 - יש להקפיד מאוד על הוראות עיצוב הקלט והפלט, בדיוק על פי הדוגמאות המצורפות.
 - בנוסף שימו לב להנחיות במסמך ה-Coding Style המפורסם באתר הקורס.
- עליכם לכתוב קוד על פי ההנחיות ולוודא שקיבלתם 100 בבדיקה האוטומטית הראשונית, וכן שהתרגיל מתקמפל ורץ על השרת המחלקתי (planet) ללא **שגיאות** או **אזהרות**.
- תרגיל שלא עומד בסטנדרטים הבסיסיים הללו יגרור **ירידה משמעותית בציון התרגיל**, בשל הטרחה שהוא מייצר בתהליך הבדיקה שלו, עד כדי ציון 0.
- להזכירכם העבודה היא אישית. "עבודה משותפת" דינה כהעתקה. התרגיל נבדק על ידי מערכת ההגשה האוטומטית גם מהבחינה הזו, ותרגיל שהועתק יגרור ציון 0 **לכל הגורמים** השותפים בהעתקה. אתם יכולים לדון בגישות לפתרון התרגיל באופן תיאורטי, אך אין לשתף קוד בשום צורה.
- בפיתוח הקוד ניתן להשתמש בכל סביבת עבודה, העיקר הוא שתדעו איך לקחת את קבצי הקוד מתוך הסביבה הזו, לבדוק אותם על שרתי האוניברסיטה ולהגיש אותם באמצעות מערכת ההגשה.
- דוגמאות לחלק מהסביבות האפשריות:
- Visual studio
 - Clion
 - Atom
 - Sublime
 - Notepad++
- שימו לב שאתם מגישים אך ורק את הקבצים המכילים את הקוד שלכם, ולא קבצים מיותרים שנוצרו על ידי סביבת העבודה. כמו כן הימנעו מהגשת קבצים/תיקיות עם שמות המכילים תווים בעברית.
- מומלץ להגיש קובץ ריק כדי לקבל במייל את הפלט המדויק שמערכת ההגשה מצפה לראות (מבחינת מחרוזות והודעות למשתמש), ולהתאים את ההדפסות בתוכנית לפלט הזה.

בתרגיל זה עליכם ליצור ולהגיש תוכנית בקובץ יחיד בשם ex_3.c
בתרגיל זה מותר להשתמש בספריות stdio.h math.h ובנוסף רק בחומר שנלמד עד התרגול בנושא מערכים.
בחלק מהסעיפים יש להשתמש ברקורסיה.

ניתן ומומלץ לעשות שימוש חוזר בפונקציות בין הסעיפים השונים כאשר זה אפשרי.

פקודת הקימפול בתרגיל זה היא:

```
gcc ex_3.c -lm -std=c99 -DNDEBUG -DSIZE=? -DLIMIT=?
```

כאשר במקום '?' יופיע מספר טבעי כלשהו, שיגדיר את גודלם של הקבועים SIZE,LIMIT בתוכנית.
שימו לב שאין להגדיר את הקבועים האלו בקוד התוכנית!

משימה 1 : הדפסת מספרים ראשוניים

במשימה זו עליכם להדפיס את כל המספרים הראשוניים עד SIZE.
למשל, עבור SIZE=100 יש להדפיס:

2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

משימה 2 : אישוש השערת גולדבאך

במשימה זו עליכם להדפיס אישוש [להשערת גולדבאך](#) שלפיה כל מספר זוגי גדול מ-2 ניתן להציג כסכום של שני מספרים ראשוניים, עבור כל המספרים הזוגיים הגדולים מ-2 שניתן להציג אותם כסכום של מספרים ראשוניים מהסעיף הקודם.

למשל עבור SIZE=10 יש להדפיס:

4 = 2 + 2
6 = 3 + 3
8 = 3 + 5
10 = 3 + 7
10 = 5 + 5
12 = 5 + 7
14 = 7 + 7

ועבור SIZE=15 יש להדפיס:

4 = 2 + 2
6 = 3 + 3
8 = 3 + 5
10 = 3 + 7
10 = 5 + 5
12 = 5 + 7
14 = 3 + 11
14 = 7 + 7
16 = 3 + 13
16 = 5 + 11
18 = 5 + 13
18 = 7 + 11
20 = 7 + 13
22 = 11 + 11
24 = 11 + 13
26 = 13 + 13

משימה 3 : מספרי ליישרל

[מספר ליישרל](#) (באנגלית: Lychrel Number) הוא מספר טבעי שאינו יוצר פלינדרום כאשר מחברים אותו עם היפוך הספרות שלו וחוזרים על הפעולה.

במשימה זו עליכם להדפיס את ההוכחה לכך שכל המספרים הטבעיים עד LIMIT (לא כולל LIMIT) אינם מספרי ליישרל. ניתן להניח שכל המספרים עד LIMIT אכן יהיו מספרי ליישרל, ושכל המספרים שעליכם להדפיס בסעיף זה ניתנים לייצוג בעזרת משתנה מטיפוס int.

עבור מספר שהוא עצמו פלינדרום, מספיק פשוט להדפיס אותו.
עבור מספר שאינו כזה, יש להדפיס את התהליך שמסתיים בפלינדרום, לדוגמה:

48:48+84=132:132+231=363

78:78+87=165:165+561=726:726+627=1353:1353+3531=4884

משימה 4: הדפסת אורך הסדרה העולה הארוכה ביותר

הערה: כל הדוגמאות לרקורסיה שראיתם עד עכשיו הגיעו יחד עם הפתרון, ולהבנה של פתרון רקורסיבי יש פחות ערך מפתרון עצמאי של הבעיה. מתישהו בהמשך הקורס (למשל במבחן) והתואר (למשל בקורסים מבני נתונים ואלגוריתמים) אתם תיתקלו בבעיות שיידרשו פתרון רקורסיבי שיהיו ככל הנראה קשות יותר מהבעיה הזו, ויהיה לכם יותר קל אם זה לא יהיה המפגש המאתגר הראשון עם בעיות מהסוג הזה. למרות זאת לא אוריד נקודות על פתרון לא רקורסיבי, אבל אני ממליץ מאוד על פתרון כזה לשאלה הזו. בהצלחה!

במשימה זו עליכם לקלוט מהמשתמש סדרת מספרים, ולהדפיס את אורכה של תת הסדרה העולה ממש הארוכה ביותר (שאינה דווקא רצופה) מתוך הסדרה שהוכנסה. לפני הכנסת הסדרה המשתמש יכניס את אורכה של הסדרה, ומובטח שמספר זה יהיה קטן מהקבוע SIZE.
למשל, עבור המספר 9 והסדרה 45,1,21,3,33,6,53,9,18 חלק מהסדרות העולות הקיימות הן:

45, 53
1, 21, 33, 53
1, 3, 33, 53
1, 3, 6, 53
1, 3, 6, 9, 18
1, 33, 53
1, 6, 53
1, 6, 9, 18
1, 53
1, 9, 18
1, 18
21, 33, 53
...

מבין כל הסדרות (כולל אלו שלא מופיעות כאן) הסדרה הארוכה ביותר היא 1,3,6,9,18 ולכן התשובה היא 5.

זרימה כללית של התוכנית

בתחילת התוכנית יש להדפיס את התפריט הבא:

- 1: primes
- 2: goldbach
- 3: lychrel
- 4: max set

לאחר מכן יש לקלוט מהמשתמש את בחירתו (ניתן להניח שיוכנס מספר שלם) ולהפעיל את הסעיף המתאים. אם הוכנס מספר שאינו 1-4 יש להדפיס:

Wrong option!

לאחר מכן יש לסיים את התוכנית.

בהמשך המסמך מצורפות חמש דוגמאות הרצה מלאות, עבור קובץ שנוצר עם הפקודה:

```
gcc ex_3.c -std=c99 -lm -DNDEBUG -DSIZE=15 -DLIMIT=30
```

בהצלחה!

1: primes
2: goldbach
3: lychrel
4: max set
1
2 3 5 7 11 13

1: primes
2: goldbach
3: lychrel
4: max set
2
 $4 = 2 + 2$
 $6 = 3 + 3$
 $8 = 3 + 5$
 $10 = 3 + 7$
 $10 = 5 + 5$
 $12 = 5 + 7$
 $14 = 3 + 11$
 $14 = 7 + 7$
 $16 = 3 + 13$
 $16 = 5 + 11$
 $18 = 5 + 13$
 $18 = 7 + 11$
 $20 = 7 + 13$
 $22 = 11 + 11$
 $24 = 11 + 13$
 $26 = 13 + 13$

1: primes
2: goldbach
3: lychrel
4: max set
3
1
2
3
4
5
6
7
8
9
 $10: 10+1=11$
11
 $12: 12+21=33$

$13: 13+31=44$
 $14: 14+41=55$
 $15: 15+51=66$
 $16: 16+61=77$
 $17: 17+71=88$
 $18: 18+81=99$
 $19: 19+91=110: 110+11=121$
 $20: 20+2=22$
 $21: 21+12=33$
22
 $23: 23+32=55$
 $24: 24+42=66$
 $25: 25+52=77$
 $26: 26+62=88$
 $27: 27+72=99$
 $28: 28+82=110: 110+11=121$
 $29: 29+92=121$

1: primes
2: goldbach
3: lychrel
4: max set
4
Enter the number of elements
9
45 1 21 3 33 6 53 9 18
5

1: primes
2: goldbach
3: lychrel
4: max set
5
Wrong option!